Students Information Management System

SOFTWARE REQUIREMENTS SPECIFICATION

Prepared by

Sitansu Subudhi (712CS1154)

NIT Rourkela

26/01/2015

Contents

1 Introduction

1.1 Document Purpose

The university student registration system has to cope with the high volume of information related to various academic and administrative activities of the students in the institute at registration time. Hence, an online student registration system needs to be developed which serves the purpose. In addition, students on campus, off campus, in-state, out of state, and out of country can easily and inexpensively take advantage of many of the services provided by the Office of the Registrar, which today require users to be on campus during business hours.

1.2 Product Scope

This software system will be a Student Management Information System for CDAC institute. It will be designed to alleviate the administration's work load by providing tools to assist in updating the students' information and other processes, which would otherwise have to be performed manually. It will also enable the administration to hold details of their students' attendance, the courses, exams and assignments matters together with details of payments. Hence, the system will be able to offer a better personalized service and meet the institute's needs while remaining easy to understand and use.

1.3 Intended Audience and Reading Suggestions

The different types of reader that this document is intended for are developers and users of the software. Moreover, this document can also be used by the administration for verifying and changing the current academic procedures.

1.4 Definitions, Acronyms and Abbreviations

IEEE : The Institute of Electrical and Electronics Engineers, Inc.

SIMS : Student Information Management System

SRS : Software Requirements Specification

SQL : Structured Query Language

Client : Intended users for the software

Use Case : A broad level diagram of the project showing a basic overview

1.5 Document Conventions

This document was prepared using the IEEE recommended practice for Software Requirements Specification.

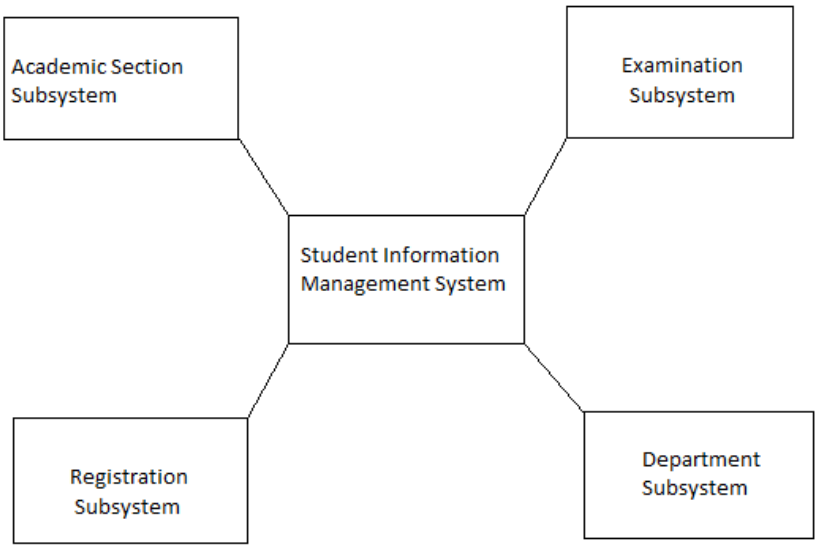
1.6 References and Acknowledgments

- en.**wikipedia**.org/**wiki**/Software_requirements_specification
- capstone.cs.ucsb.edu/cs189a/support/**SRS**-**template**.doc
- http://www.academia.edu
- http://www.scribd.com
- http://www.slideshare.net
- https://www.google.co.in

2 Overall Description

2.1 Product Perspective

SIMS is meant to serve as a common platform where management of everyday academic tasks can be carried out conveniently. The goal is to develop a user friendly interface between the different subsystems and to promote academic networking among the users. These subsystems include department, academic section, registration, examination etc.



2.2 Product Functionality

The proposed system will:
-Enable administration register, delete and find student records.
-Store details about the students actually studying in the institute, including their personal details, whether they are full-time or part-time students, a progress sheet where information about their exams marks will be will be recorded and
-Include all essential data on the courses offered; the modules and the respective lecturers, the different batches of the courses will also be available,
-Keep payments follow ups up-to-date.
-Allow lecturers to record attendance of the students.
-Enable the lecturers to insert progress of students as their assignments marks.
-Display exams, date and time scheduled, assignments title and markings.
-Allow program officers to validate students' results.
-Also included a list of classes scheduled created by the administration.
-Produce reports as:
. List of exams schedules
. List of exam results
. List of attendance
. List of payments follow-up
. List of results statistics
. List of the assignments marks
. List of batches
. List of classes

Eventually, the new system will enable the lecturers and the administration to save time because the proposed system will consist of a shared database to allow all the people involved to access it so as to insert, retrieve, delete and view all the information needed. There will be individual input of the information at a specific time of the day so as all the information is updated as efficiently as possible.

2.3 Users and Characteristics

The users include administrator, student, and faculty. The users are assumed to have basic knowledge of the computers and Internet browsing. The administrators of the system to have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to disk crashes, power failures and other catastrophes to maintain the system.

2.4 Operating Environment

The SIMS is a website and can work perfectly in any internet browser, preferably if its Google Chrome or Mozilla Firefox having all the latest features. Any operating system can support the functioning of this website.

2.5 Design and Implementation Constraints

The information of all users must be stored in a database that is accessible by the website.

· MS SQL Server will be used as SQL engine and database.
· The SIMS should be functional 24 hours a day.
· Users may access SIMS from any computer that has Internet browsing capabilities and an Internet connection.
· Users must have their correct usernames and passwords to enter into their online accounts and do actions.

2.6 Assumptions and Dependencies

The staff is expected to be Internet literate and be able to use the email with attachments. They are also expected to be Windows literate and to be able to use button, pull-down menus, and similar tools.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Each level of user will have its own interface and privilege to manage and modify the project information such as supervisor able to monitor/manage his student progress and make comment on it, student can change his detail, view the progress, submit project idea. The System should provide a feedback form for all users to give comments or asking questions. It should provide a FAQ to minimize the workload of system administrator.
For initial registration of the users, the system administrator needs to create a new entry having attributes of the individual such as his name, identification number and other required data as per the requirements. The database then needs to be updated with the new entry.

3.1.2 Hardware Interfaces

Server Side:
The web application will be hosted on one of the department's Linux servers and connecting to one of the school MySQL Database server. The web server is listening on the web standard port, port 80.

Client Side:
The system is a web based application; clients are requiring using a modern web browser such as Mozilla Firebox, Internet Explorer, and Google Chrome. The computer must have an Internet connection in order to be able to access the system.

3.1.3 Software Interfaces

Server Side:
A Web server will accept all requests from the client and forward specific requests. A development database will be hosted locally (using MySQL); the production database is hosted centrally (using Oracle).

Client Side:
An OS is capable of running a modern web browser which supports HTML version 3.2 or higher.

3.1.4 Communications Interfaces

Server communications for interaction between the administrator and logged in users is required. The HTTP or HTTPS protocol(s) will be used to facilitate communications between the client and server.

3.2 Behaviour Requirements

3.2.1 Use Case View

Login

The login credentials decide whether the user is an administrator or any of other users. This is used to validate against the users database and assign users to groups. Only if the username and password is correct, does the user get access to the system.

Create New Users

Only the administrator has rights to create a new user. Here the basic user details like his first name, last name, username and password are entered.

Get Access Rights

This is very important module, which decides what areas of the system are accessible to the users depending on their login credentials. Not all areas of the system are accessible to all users.

Access Database

Once user's rights are determined access to the system database is given.

View listed information

All the information present in the database is available to the user in a friendly format. Menu driven pages will help the user to get proper content.

Update Information

Users with updating rights can update their information and save it back to the system.

Add New Records

Users with addition rights can add new records to their names. Administrators have rights to add records to any user of the system.

4 System Features

4.1 Registration

4.1.1 Description and Priority

Students can choose to do their pre-registration, de-registration or view current or previous registration/de-registration reports.

4.1.2 Stimulus/Response Sequences

- User enters his/her roll no as login ID and enters the password.
- User Clicks on 'Pre-Registration', and then on 'Autumn/Spring' for doing Pre-Registration.
- Enters the required information (preferred courses and optional electives list) and submits.
- User Clicks on 'De-Registration', and then on 'Autumn/Spring'.
- Selects the course to de-register from the available choices.
- Submits the de-registration form.
- User Clicks on 'Reports', and then on 'View Registration/De-registration details'.
- User views the necessary details of registration/de-registration.
- Logs out.

4.1.3 Functional Requirements

4.1.3.1: PreRegister() : Student can pre-register for the courses of the next semester here.

Input: User enters password and log-in and click on "Pre-Registration".

Output: If within the last date, System shows all pre-registration options(courses available, professional electives, Open electives, etc) and on completion of form fill-up, saves the information to the student database and shows the filled-up options. It's re-editable until the last date. Else shows "last date is over".

REQ-1: Login Id (Institute Roll Number)
REQ-2: Password

4.1.3.2: DeRegister() : If the student wants to reduce the credits for the current semester, he can do so by de-registering some or all of the courses here.

Input: User logged-in and click on "De-Registration".

Output: If within the last date, System shows all de-registration options(courses available, professional electives, Open electives, etc) and on completion of form fill-up, saves the information to the student database and shows the filled-up options. It's re-editable until the last date. Else shows "last date is over".

REQ-1: Login Id (Institute Roll Number)
REQ-2: Password

4.1.3.3: RegistrationReport() : Student finds the details of the registration report here.

Input: User click on "View Registration Details".

Output: System loads a form with details such as 'Academic Year', 'Semester', 'Subject Code', 'Subject Name', and 'Credits'. If the user changes

the 'Academic Year' or the 'Semester', the corresponding relevant information is shown.

4.2 Department

4.2.1 Description and Priority

In this department specific subsystem, the registered faculty are assigned the outlined course and question papers and assignments are uploaded.

4.2.2 Stimulus/Response Sequences

- Department faculty enters the login ID and the respective password.
- Faculty clicks on Allotted Course.
- Views the allotted course code, name, type and credits of the course.
- Faculty clicks on a particular course code from the above.
- Views the number of registered students, details of the students including roll no., name, TA marks, attendance and assignments submitted.
- Uploads the mark sheet, attendance sheet, or assignment.
- Faculty clicks on 'Upload Question Paper'.
- Uploads a Question Paper as per the maximum marks for the type of exam.
- Makes any required changes.
- Logs out.

4.2.3 Functional Requirements

4.2.3.1: AllotedCourses() : Courses allotted to the respective faculty is seen here.

Input: User logged-in and click on "Allotted Courses".

Output: Shows the allotted courses (code, name, credits) to the faculty, else shows "No Courses Allotted". Each course is hyperlinked to show number of registered students, details of the students including roll no., name, TA marks, attendance and assignments submitted.

REQ-1: Login Id (Institute Roll Number)
REQ-2: Faculty Password

4.2.3.2: QuesUpload() : Faculty uploads the question paper required for the mid or end semester exam.

Input: User logged-in and click on "Upload Question Paper".

Output: The Question Paper is uploaded and then returns to home page. The document can be edited and submitted multiple number of times.

REQ-1: Login Id (Faculty ID)
REQ-2: Course Id.

4.3 Student

4.3.1 Description and Priority

This is a student specific subsystem and one of the most important subsystems. This module shows the student his personal information from the stored records and his faculty advisor and tuition and mess fees.

4.3.2 Stimulus/Response Sequences

- User Clicks on 'Personal Details'.

- Views the initially stored information from the Master Database like name, parents' name, date of birth, Blood Group, AIEEE Rank, Category, Date of Joining, Batch, Faculty Advisor, and bank account details.
- Clicks on "Fees".
- Shows the tuition fees, library fine, books fair fees, medical fees, mess fees, if updated respectively by the academic section and the Chief Warden.
- Logs out.

4.3.3 Functional Requirements

4.3.3.1: StudDetails() : Student personal details found here.

Input: User logged-in and click on "Personal Details".

Output: Shows the information available since the first registration from the Master Database like name, parents' name, date of birth, Blood Group, AIEEE Rank, Category, Date of Joining, Batch, Faculty Advisor, and bank account details.

REQ-1: Login Id (Institute Roll Number)

4.3.3.2: ViewFees() : Amount of fees payed.

Input: User logged-in and click on "Fees".

Output: The Semester fees are shown, including mess fess, library fine, books fair fees, medical fees, tuition fees and others.

REQ-1: Login Id (Institute Roll Number)
REQ-2: Fees set by academics
REQ-3: Library Fines
REQ-4: Hall Dues

4.4 Academic Section

4.4.1 Description and Priority

This is also another important functionality where the authorities decide and implement the various rules related to grade backs and debarment, change in course structure, and change in faculty. Moreover, the attendances for each course are shown.

4.4.2 Stimulus/Response Sequences

- Student Clicks on 'Attendance'.
- Views the attendance details (including grade back, debars and UR)
- Clicks on 'Attendance Rules'
- Views the currently active institute rules on attendance system.
- Clicks on 'Change in Course Structure'.
- Views any changes in the structure of the registered courses.
- Clicks on 'Change in Faculty'.
- Views any changes in the assigned faculty under the registered courses,
- Logs out.

4.4.3 Functional Requirements

4.4.3.1: Attendance() : Complete attendance record of student found here.

Input: User logged-in and click on "Attendance". User must be registered.

Output: Shows the attendance details for the registered courses for the user for each days of the month, and any grade backs or debars. Shows biometric attendance if for Research Scholars.

REQ-1: Login Id (Institute Roll Number)
REQ-2: Uploaded attendance
REQ-3: Biometric Attendance Data
REQ-4: Current grade back rules

4.4.3.2: AttendanceRules() : The rules regarding debarring in subjects or gradeback rules found here.

Input: User logged-in and click on "Attendance Rules".

Output: Displays the currently prevalent attendance rules in the institute, and the last modified date.

4.4.3.3: CourseChange() : Changing of a particular core or elective subject or its respective syllabus.

Input: User logged-in and click on "Change in Course Structure".

Output: Displays any changes in the structure of the registered courses and the date, else shows 'no change'.

REQ-1: Login Id (Institute Roll Number)
REQ-2: Registered Courses

4.5 Examination

4.5.1 Description and Priority

Another part of the SIMS; this subsystem deals with examinations. Shows the next exam sitting chart and examination results.

4.5.2 Stimulus/Response Sequences

- Student Clicks on 'Examination Sitting Chart'.
- Views the sitting chart of the approaching exam in the registered courses.
- Clicks on a particular course code
- Views the expanded sitting chart for the particular exam, including row and column numbers, room number, date and day.
- Clicks on 'Examination Results'
- Views Midsem/ Endsem marks for all the registered courses semester-wise.
- Clicks on "grade card".
- Views the updated grade card.
- Logs out.

4.5.3 Functional Requirements

4.5.3.1: ExamChart() : Exam sitting chart available here.

Input: User logged-in and click on "Examination Sitting Chart".

Output: Shows the Examination Sitting Arrangement for the coming exams in a table format, divided into rows and columns. Date and Shift of exam, including subject code, room no are also available.

REQ-1: Login Id (Institute Roll Number)
REQ-2: Exam sitting chart prepared by the Exam Section.

4.5.3.2: ExamRes() : Printing of results after evaluation.

Input: User logged-in and click on "Examination results".

Output: Displays the Midsem/ Endsem marks for all the registered courses semester-wise.

4.5.3.3: GradeCard() : Grade sheet of the respective students.

Input: User logged-in and click on "Grade Card".

Output: Displays the updated grade card. Option is made available to print/save a copy.

REQ-1: Login Id (Institute Roll Number)
REQ-2: ExamRes()

5 Other Non-functional Requirements

5.1 Performance Requirements

The website would be functional for 24 hours a day to enable user interaction at any point of time.

5.2 Safety Requirements

The database should be carefully maintained by the administrator any loss may lead to chaos.

Prevention of Fake Student/Professor IDs.

5.3 Security Requirements

The Administrator password must be highly confidential.
The User ID must be confidential.

5.4 Software Quality Attributes

Adaptability: New changes can be easily accommodated

Availability: 24*7 availability of services.

Interoperability: Each interface makes use of information provided by the other.

Flexibility: Provides highly flexible services to the users.

Portability: Not dependent on particular OS.